# You Are What You Like! Information Leakage Through Users' Interests

Abdelberi Chaabane, Gergely Acs, Mohamed Ali Kaafar
INRIA France
{chaabane, gergely.acs, kaafar}@inrialpes.fr

## Abstract

*Suppose that a Facebook user, whose age is hidden or missing, likes Britney Spears. Can you guess his/her age? Knowing that most Britney fans are teenagers, it is fairly easy for humans to answer this question. Interests (or "likes") of users is one of the highly-available on-line information. In this paper, we show how these seemingly harmless interests (e.g., music interests) can leak privacy-sensitive information about users. In particular, we infer their undisclosed (private) attributes using the public attributes of other users sharing similar interests. In order to compare user-defined interest names, we extract their semantics using an ontologized version of Wikipedia and measure their similarity by applying a statistical learning method. Besides self-declared interests in music, our technique does not rely on any further information about users such as friend relationships or group belongings. Our experiments, based on more than 104K public profiles collected from Facebook and more than 2000 private profiles provided by volunteers, show that our inference technique efficiently predicts attributes that are very often hidden by users. To the best of our knowledge, this is the first time that user interests are used for profiling, and more generally, semantics-driven inference of private data is addressed.*

## 1. Introduction

Among the vast amount of personal information, user interests or *likes* (using the terminology of Facebook) is one of the highly-available public information on On-line Social Networks (OSNs). Our measurements show that 57% of about half million Facebook user profiles that we collected *publicly* reveal at least one interest amongst different categories. This wealth of information shows that the majority of users consider this information harmless to their privacy as they do not see any correlation between their interests and their private data. Nonetheless, interests, if augmented with semantic knowledge, may leak information about its owner and thus lead to privacy breach. For example, consider an unknown Facebook user who has an interest "Eenie Meenie". In addition, there are many female teenager users who have interests such as "My World 2.0" and "Justin Bieber". It is easy to predict that the unknown user is probably also a female teenager: "Eenie Meenie" is a song of "Justin Bieber" on his album "My World 2.0", and most Facebook users who have these interests are female teenagers. This example illustrates the two main components of our approach: (1) deriving *semantic correlation* between words (e.g., "My World 2.0", "Eenie Meenie", and "Justin Bieber") in order to link users sharing similar interests, and (2) deriving statistics about these users (e.g., Justin Bieber fans) by analyzing their public Facebook profiles. To the best of our knowledge, the possibility of this information leakage and the automation of such inference have never been considered so far. We believe that this lack of exploitation is due to several challenges to extract useful information from interest names and descriptions.

First, many interests are *ambiguous*. In fact, they are short sentences (or even one word) that deal with a concept. Without a semantic definition of this concept, the interest is equivocal. For example, if a user includes "My World 2.0" in her Art/Entertainment interests, one can imply that this user is likely to be interested in pop as a genre of music. Without a knowledge of what "My World 2.0" is, the information about such an interest is hidden, and hence unexploited.

Second, drawing *semantic link* between different interests is difficult. For example, if a user includes in her public profile "My World 2.0" and another user chooses the interest "I love Justin Bieber", then clearly, these two users are among the Justin Bieber funs. However, at a large scale, automating interest linkage may not be possible without semantic knowledge.

Finally, interests are *user-generated*, and as such, very *heterogeneous* items as opposed to marketers' classified items (e.g., in Amazon, Imdb, etc.). This is due to the fact that OSNs do not have any control on how the descriptions and titles of interests are constructed. As a result, interest descriptions as provided by users are often incorrect, misleading or altogether missing. It is therefore very diffi-

cult to extract useful information from interests and classify them from the user-generated descriptions. Particularly, interests that are harvested from user profiles are different in nature, ranging from official homepage links or ad-hoc created groups to user instantaneous input. In addition, interest descriptions, as shown on public profiles, either have coarse granularity (i.e., high level descriptions of classes of interests such as "Music", "Movies", "Books", etc.), or they are too fine-grained to be exploited (e.g., referring to the name of a singer/music band, or to the title of a recent movie, etc.). Finding a source of knowledge encompassing this huge variety of concepts is challenging.

Therefore, *linking users sharing semantically related interests* is the pivot of our approach. The main goal of our work is to show how seemingly harmless information such as interests, if augmented with semantic knowledge, can leak private information. As a demonstration, we will show that *solely based* on what users reveal as their music interests, we can successfully infer hidden information with more than 70% of correct guesses for some attributes in Facebook. Furthermore, as opposed to previous works [18, 27, 22], our technique *does not need further information*, such as friend relationships or group belongings.

### Technical Roadmap

Our objective is to find out interest similarities between users, even though these similarities might not be clearly observed from their interests. We extract semantic links between their seemingly unrelated interest names using the Latent Dirichlet Allocation (LDA) generative model [6]. The idea behind LDA is to learn the underlying (semantic) relationship between different interests, and classify them into "unobserved" groups (called Interest Topics). The output of LDA is the probabilities that an interest name belongs to each of these topics.

To identify latent (semantic) relations between different interests, LDA needs a broader semantic description of each interest than simply their short names. For instance, LDA cannot reveal semantic relations between interests "Eenie Meenie" and "My World 2.0" using only these names unless they are augmented with some text describing their semantics. Informally, we create a document about "Eenie Meenie" and another about "My World 2.0" that contain their semantic description and then let LDA identify the common topics of these documents. These documents are called Interest Descriptions. In order to draw semantic knowledge from the vast corpus of users' interests, we leverage on the ontologized version of Wikipedia. An interest description, according to our Wikipedia usage, is the parent categories of the most likely article that describes the interest. These represent broader topics organizing this interest. For instance, there is a single Wikipedia article about "Eenie Meenie" which belongs to category "Justin Bieber songs" (among others). In addition, there is another article about "My World 2.0" that belongs to category "Justin Bieber albums". Therefore, the descriptions of interests "Eenie Meenie" and "My World 2.0" will contain "Justin Bieber songs" and "Justin Bieber albums", respectively, and LDA can create a topic representing Justin Bieber which connects the two interests. An interesting feature of this method is the ability to enrich the user's interests from, say a single item, to a collection of related categories, and hence draw a broader picture of the semantics behind the interest of the user. We used two sets of 104K public Facebook profiles and 2000 private profiles to derive the topics of all the collected interests.

Knowing each user's interests and the probabilities that these interests belong to the identified topics, we compute the likelihood of users are interested in these topics. Our intuition is that users who are interested roughly in the same topics with "similar" likelihood (called interest neighbors) have also similar personal profile data. Hence, to infer a specific user's hidden attribute in his profile, we identify his interest neighbors who publicly reveal this attribute in their profile. Then, we guess the hidden value from the neighbors' (public) attribute values.

We postulate and verify that interest-based similarities between users, and in particular their music preferences, is a good predictor of hidden information. As long as users are revealing their music interests, we show that sensitive attributes such as Gender, Age, Relationship status and Country-level locations can be inferred with high accuracy.

**Organization** We describe our attacker model in Section 2. Section 3 presents related work and show the main differences between our approach and previous works. Our algorithm is detailed in Section 4 and both of our datasets are described in Section 5. Section VI is devoted to present our inference results. We discuss some limitations of our approach and present future works in Section 7 and finally we conclude.

## 2. Attacker Model

Before defining our attacker model, we describe user profiles as implemented by Facebook.

Facebook implements a user profile as a collection of personal data called attributes, which describe the user. These attributes can be binary such as Gender or multi-values such as Age. The availability of these attributes obeys to a set of privacy-settings rules. Depending on these privacy settings, which are set by the user, information can be revealed exclusively to the social links established on

OSN (e.g., friends in Facebook[1]) or partially (e.g., to friends of friends) or publicly (i.e., to everyone). In this paper, we demonstrate the information leakage through users' interests by inferring the private attributes of a user. We consider two binary attributes (Gender: male/female and Relationship status: married/single) and two multi-valued attributes (Country-level location and Age).

As opposed to previous works [18, 27], we consider an attacker that *only* has access to self-declared, publicly available music interests. Hence, the attacker can be *anyone* who can collect the Facebook public profile of a targeted user. In fact, earlier attacks considered a dataset crawled from a specific community such as a university. Hence, the crawler being part of this community had access to attributes that are only visible to friends which impacts data availability. Indeed, as we will show in Section 5, data availability is different whether we deal with public data (data disclosed to anyone) or private data (data disclosed to friends only). Thus our attacker is more general compared to [18, 27], since it relies only on public information.

This characteristic allows us to draw a broader attacker. For example, our technique can be used for the purpose of profiling to deliver targeted ads. Advertisers could automatically build user profiles with high accuracy and minimum effort, with or *without* the consent of the users. Spammers could gather information across the web to send extremely targeted spam (e.g., by including specific information related to the location or age of the targeted user).

## 3. Related Work

Most papers have considered two main privacy problems in OSNs: inferring private attributes and de-anonymizing users. Most of these works used the information of friendships or group belongings in order to achieve these goals. By contrast, our approach *only* relies on users' interests. In particular, instead of using link based classification algorithms [11] and/or mixing multiple user attributes to improve inference accuracy, we provide a new approach based on semantic knowledge in order to demonstrate information leakage through user interests. Moreover, all previous works relied on private datasets (e.g., dataset of a private community such as a university), and hence assumed a different attacker model than ours (see Section 2 for details). We also leverage knowledge from the area of Personalizing Retrieval in order to link users sharing similar interests.

**Private Attribute Inference** Zheleva and Getoor [27] were the first to study the impact of friends' attributes on the privacy of a user. They tried to infer private user attributes based on the groups the users belong to. For that purpose, they compared the inference accuracy of different link-based classification algorithms. Although their approach provides good results for some OSNs such as Flicker, they admit that it is not suitable to Facebook especially with multi-valued attributes such as political views. Moreover, they made the assumption that at least *50%* of a user's friends reveal the private attribute. However, our experiments show that this is not realistic in our attacker model, since users tend to (massively) hide their attributes from public access (see Section 5). For instance, only 18% of users on Facebook disclose their relationship status and less than 2% disclose their birth date.

In [13], authors built a Bayes network from links extracted from a social network. Although they crawled a real OSN (LiveJournal) they used hypothetical attributes to analyze their learning algorithm. A further step was taken by [18] who proposed a modified Naive Bayes classifier that infers political affiliation (i.e., a binary value: liberal or conservative) based on user attributes, user links or both. Besides a different attacker model, we do not use the combination of multiple attributes to infer the missing one (i.e., we only use music interests).

Rather than relying on self declared or existing graphs, Mislove et al. [22] built "virtual" communities based on a metric called *Normalized conductance*. However, community-based inference is data dependent because the detected community may not correlate with the attribute to be inferred. Indeed, [25] provided an in depth study of community detection algorithms for social networks. After comparing the results of 100 different social graphs (provided by Facebook), they concluded that a common attribute of a community is good predictor only in certain social graphs (e.g., according to [25], the communities in the MIT male network are dominated by residence, but it is not the case for female networks).

**De-anonymizing Users** In [5], the authors considered an anonymized network composed of nodes (users) and edges (social links) where the attacker aims to identify a "targeted" user. Another problem was considered by [26], where a targeted user visiting a hostile website was de-anonymized using his group belongings (stolen from his web-history). The main idea behind both attacks is that the group membership of a user is in general sufficient to identify him/her. De-anonymization of users, considered by these works, is an orthogonal privacy risk to attribute inference.

**Personalizing Retrieval** Our work shares techniques with the area of personalizing retrieval where the goal is to build personalized services to users. This can be derived from the user "taste" or by interpreting his social interactions. This is an active research domain and a broad range

[1] Facebook has recently added a feature to split friends into sublists in order to make some attributes accessible to a chosen subset of friends.

of problems were resolved and used in e-commerce, recommendation, collaborative filtering and similar. This knowledge extraction entails the analysis of a large text corpora from which one can derive a statistical model that explains latent interactions between the documents. Latent semantic analysis techniques provide an efficient way to extract underlying topics and cluster documents [14, 9]. Latent Dirichlet Allocation (LDA) [6] has been extended by Zhang et al. [7] to identify communities in the Orkut social network. The model was successfully used to recommend new groups to users. In addition, Zheleva et al. [28] used an adapted LDA model to derive music taste from listening activities of users in order to identify songs related to a specific taste and the listeners who share the same taste.

Similarly to these works, we also use LDA to capture the interest topics of users but instead of recommending content, our goal is to link users sharing *semantically-related* interests to demonstrate information leakage.

# 4. From Interest Names to Attribute Inference

## 4.1. Overview

While a human can easily capture the semantics behind different interest names (titles or short descriptions), this task cannot be easily automated. In this section, we present how we can extract meaningful knowledge from users' interests and then classify them for the purpose of attribute inference.

Our technique consists of four main steps as illustrated by Figure 1:

1. Creating Interest Descriptions: Interest descriptions are the user-specified interest names augmented with semantically related words which are mined from the Wikipedia ontology.

2. Extracting semantic correlation between interest descriptions using Latent Dirichlet Allocation (LDA). The output represents a set of topics containing semantically related concepts.

3. Computing Interest Feature Vectors (IFV). Based on the discovered topics, LDA also computes the probability that an interest $I$ belongs to $Topic_i$ for all $I$ and $i$ (Step 3a). Then, we derive the IFV of each user (Step 3b) which quantifies the interest of a user in each topic.

4. Computing the neighbors of each user in the feature space (i.e., whose IFVs are similar in the feature space) to discover similar users, and exploiting this neighborhood to infer hidden attributes.

## 4.2. Step 1: Augmenting Interests

Interest names (shortly interests) extracted from user profiles can be single words, phrases, and also complex sentences. These text fragments are usually insufficient to characterize the interest *topics* of the user. Indeed, most statistical learning methods, such as LDA, need a deeper description of a given document (i.e., interest) in order to identify the semantic correlation inside a text corpora (i.e., set of interests). Moreover, the diversity and heterogeneity of these interests make their description a difficult task. For instance, two different interests such as "AC/DC" and "I love Angus Young" refer to the same band. However, these strings on their own provide insufficient information to reveal this semantic correlation. To augment interest names with further content that helps LDA to identify their common topics, we use an ontology, which provides structured knowledge about any unstructured fragment of text (i.e., interest names).

### 4.2.1 Wikipedia as an Ontology

Although there are several available ontologies [10, 3], we use the ontologized version of Wikipedia, the most up-to-date and largest reference of human knowledge in the world. Wikipedia represents a huge, constantly evolving collection of manually defined concepts and semantic relations, which are sufficient to cover most interest names. Moreover, Wikipedia is multilingual which allows the augmentation of non-english interest names. We used the Wikipedia Miner Toolkit [21] to create the ontologized version of Wikipedia from a dump made on January, 2011 with a size of 27 Gb.

Wikipedia includes *articles* and *categories*. Each article describes a single concept or topic, and almost all Wikipedia's articles are organized within one or more categories, which can be mined for broader (more general) semantic meaning. AC/DC, for example, belongs to the categories Australian hard rock musical groups, Hard rock musical groups, Blues-rock groups etc. All of Wikipedia's categories descend from a single root called *Fundamental*. The distance between a particular category and this root measures the category's generality or specificity. For instance, AC/DC is in depth 5, while its parent categories are in depth 4 which means they are more general and closer to the root. All articles contain various hyper links pointing to further (semantically related) articles. For example, the article about Angus Young contains links to articles AC/DC, musician, duckwalk, etc. The anchor texts used within these links have particular importance as they can help with disambiguation and eventually identifying the most related article to a given search term: e.g., if majority of the "duckwalk" links (i.e., their anchor texts contain string "duck-

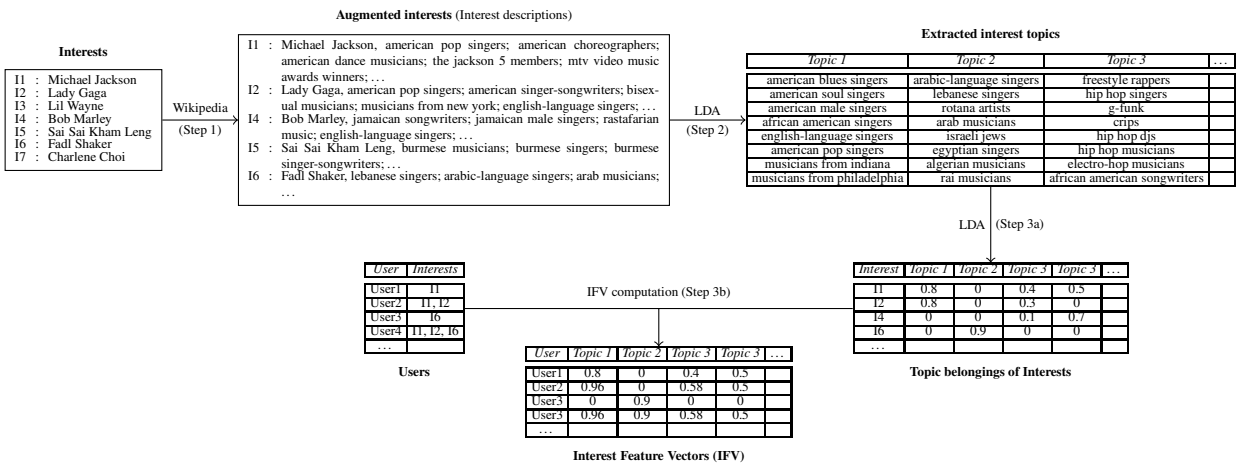


Figure 1: Computing interest feature vectors. First, we extract interest names and augment them using Wikipedia (Step 1). Then, we compute correlation between augmented interests and generate topics (Step 2) using LDA. Finally, we compute the IFV of each user (Step 3).

walk") is pointing to Chuck Berry and only a few of them to the bird Duck, then with high probability the search term "duckwalk" refers to Chuck Berry (a dancing style performed by Chuck Berry). Indeed, the toolkit uses this approach to search for the most related article to a search string; first, the anchor texts of the links made to an article are used to index all articles. Then, the article which has the most links containing the search term as the anchor text is defined to be the most related article.

#### 4.2.2 Interest Description

The description of an interest is the collection of the parent categories of its most related Wikipedia article (more precisely, the collection of the name of these categories). To create such descriptions, we first searched for the Wikipedia article that is most related to a given interest name using the toolkit. The search vocabulary is extensive (5 million or more terms and phrases), and encodes both synonymy and polysemy. The search returns an article or set of articles that could refer to the given interest. If a list is returned, we select the article that is most likely related to the interest name as described above. Afterwards, we gather all the parent categories of the most related article which constitute the description of the interest. For example, in Figure 1, *User3* has interest "Fadl Shaker". Searching for "Fadl Shaker" in Wikipedia, we obtain a single article which has parent categories "Arab musicians", "Arabic-language singers" and "Lebanese male singers". These strings altogether (with "Fadl Shaker") give the description of this interest.

### 4.3. Step 2: Extracting Semantic Correlation

To identify semantic correlations between interest descriptions, we use Latent Dirichlet Allocation (LDA) [6]. LDA captures statistical properties of text documents in a discrete dataset and represents each document in terms of the underlying topics. More specifically, having a text corpora consisting of $N$ documents (i.e., $N$ interest descriptions), each document is modeled as a mixture of latent topics (interest topics). A topic represents a cluster of words that tend to co-occur with a high probability within the topic. For example, in Figure 1, "American soul singers" and "American blues singers" often co-occur and thus belong to the same topic ($Topic_1$). However, we do not expect to find "Arab musicians" in the same context, and thus, it belongs to another topic ($Topic_2$). Note that the topics are created by LDA and they are not named. Through characterizing the statistical relations among words and documents, LDA can estimate the probability that a given document is about a given topic where the number of all topics is denoted by $k$ and is a parameter of the LDA model.

More precisely, LDA models our collection of interest descriptions as follows. The topics of an interest description are described by a discrete (i.e., categorical) random variable $\mathcal{M}(\phi)$ with parameter $\phi$ which is in turn drawn from a Dirichlet distribution $\mathcal{D}(\alpha)$ for each description, where both $\phi$ and $\alpha$ are parameter vectors with a size of $k$. In addition, each topic $z$ out of the $k$ has a discrete distribution $\mathcal{M}(\beta_z)$ on the whole vocabulary. The generative process for each interest description has the following steps:

1. Sample $\phi$ from $\mathcal{D}(\alpha)$.

2. For each word $w_i$ of the description:

(a) Sample a topic $z_i$ from $\mathcal{M}(\phi)$.

(b) Sample a word $w_i$ from $\mathcal{M}(\beta_{z_i})$.

Note that $\alpha$ and $B = \cup_z\{\beta_z\}$ are corpus-level parameters, while $\phi$ is a document-level parameter (i.e., it is sampled once for each interest description). Given the parameters $\alpha$ and $B$, the joint probability distribution of an interest topic mixture $\phi$, a set of words $W$, and a set of $k$ topics $Z$ for a description is

$$p(\phi, Z, W|\alpha, B) = p(\phi|\alpha)\prod_{\forall i} p(z_i|\phi)p(w_i|\beta_{z_i}) \quad (1)$$

The observable variable is $W$ (i.e., the set of words in the interest descriptions) while $\alpha$, $B$, and $\phi$ are latent variables. Equation (1) describes a parametric empirical Bayes model, where we can estimate the parameters using Bayes inference. In this work, we used collapsed Gibbs sampling [19] to recover the posterior marginal distribution of $\phi$ for each interest description. Recall that $\phi$ is a vector, i.e., $\phi_i$ is the probability that the interest description belongs to $Topic_i$.

### 4.4. Step 3: Interest Feature Vector (IFV) Extraction

The probability that a user is interested in $Topic_i$ is the probability that his interest descriptions belong to $Topic_i$. Let $V$ denote a user's interest feature vector, $\mathbb{I}$ is the set of his interest descriptions, and $\phi_i^I$ is the probability that interest description $I$ belongs to $Topic_i$. Then, for all $1 \leq i \leq k$,

$$V_i = 1 - \prod_{\forall I \in \mathbb{I}} (1 - \phi_i^I)$$

is the probability that the user is interested in $Topic_i$.

For instance, in Figure 1, *User4* has interests "Lady Gaga", "Michael Jackson", and "Fadl Shaker". The probability that *User4* belongs to $Topic_1$, which represents American singers, is the probability that at least one of these interests belongs to $Topic_1$. This equals $1 - ((1 - 0.8)(1 - 0.8)) = 0.96$.

### 4.5. Step 4: Inference

#### 4.5.1 Neighbors Computation

Observe that an IFV uniquely defines the interest of an individual in a $k$-dimensional feature space. Defining an appropriate distance measure in this space, we can quantify the similarity between the interests of any two users. This allows the identification of users who share similar interests, and likely have correlated profile data that can be used to infer their hidden profile data.

We use a chi-squared distance metric. In particular, the correlation distance $d_{V,W}$ between two IFV vectors $V$ and $W$ is

$$d_{V,W} = \sum_{i=1}^{k} \frac{(V_i - W_i)^2}{(V_i + W_i)}$$

In [23], authors showed that the chi-squared distance gives better results when dealing with vectors of probabilities than others. Indeed, we conducted several tests with different other distance metrics: Euclidean, Manhattan and Kullback-Leibler, and results show that the chi-squared distance outperforms all of them.

Using the above metric, we can compute the $\ell$ nearest neighbors of a user $u$ (i.e., the users who are the closest to $u$ in the interest feature space). A naive approach is to compute all $M^2/2$ pairwise distances, where $M$ is the number of all users, and then to find the $\ell$ closest ones for each user. However, it becomes impractical for large values of $M$ and $k$. A more efficient approach using $k$-$d$ tree is taken. The main motivation behind $k$-$d$ trees is that the tree can be constructed efficiently (with complexity $O(M \log_2 M)$), then saved and used afterwards to compute the closest neighbor of any user with a worst case computation of $O(k \cdot M^{1-1/k})$.

#### 4.5.2 Inference

We can infer a user's hidden profile attribute $x$ from that of its $\ell$ nearest neighbors: first, we select the $\ell$ nearest neighbors out of all whose attribute $x$ is defined and public. Then, we do *majority voting* for the hidden value (i.e., we select the attribute value which the most users out of the $\ell$ nearest neighbor have). If more than one attribute value has the maximal number of votes, we randomly choose one.

For instance, suppose that we want to infer *User4*'s country-level location in Figure 1, and *User4* has 5 nearest neighbors (who publish their locations) because all of them are interested in $Topic_2$ with high probability (e.g., they like "Fadl Shaker"). If 3 out of these 5 are from Egypt and the others are from Lebanon then our guess for *User4*'s location is Egypt.

Although there are multiple techniques besides majority voting to derive the hidden attribute value, we will show in Section 6.2 that, surprisingly, even this simple technique results in remarkable inference accuracy.

## 5. Dataset Description

For the purpose of our study, we collected two profile datasets from Facebook. The first is composed of Facebook profiles that we crawled and which we accessed as "everyone" (see Section 5.1). The second is a set of more than 4000 private profiles that we collected from volunteers using a Facebook application (see Section 5.2). Next, we describe our methodology used to collect these datasets. We

also present the technical challenges that we encountered while crawling Facebook. Finally, we describe the characteristics of our datasets.

### 5.1. Crawling Public Facebook Profiles

Crawling a social network is challenging due to several reasons. One main concern is to avoid sampling biases. A previous work [15] has shown that the best approach to avoid sampling bias is a so called True Uniform Sampling (UNI) of user identifiers (ID). UNI consists in generating a random 32-bits ID and then crawling the corresponding user profile in Facebook. This technique has a major drawback in practice: most of the generated IDs are likely to be unassigned, and thus not associated with any profile (only 16% of the 32-bits space is used). Hence, the crawler would quickly become very resource-consuming because a large number of requests would be unsuccessful. In our case, inspired by the conclusions in [15], and avoiding sampling bias that might be introduced by different social graph crawls (e.g. Breadth-First Search), we follow a simple, yet efficient two-steps crawling methodology as an alternative to UNI.

First, we randomly crawled a large fraction of the Facebook Public directory[3]. As a result, a total of 100 Million (and 120 thousands) *URL*s of searchable Facebook profiles were collected (without profile data). This technique allows to avoid the random generation of user identifiers while uniformly (independently from the social graph properties) collecting *existing* user identifiers.

Second, from this list of candidate URLs of profiles, we crawled a set of randomly selected 494 392 profiles out of the 100 millions. The crawled dataset is called RawProfiles.

Finally, the entire RawProfiles dataset was sanitized to fit our validation purposes. Two restrictions were considered: (1) non Latin-written profiles were filtered out from the dataset and (2) only profiles with at least one music interest with its corresponding Wikipedia description were kept. Therefore, we obtained a set of 104 401 profiles. This data set, called PubProfiles, is then used as an input of our inference algorithm (see details in Section 4.4).

#### Technical challenges

As noted above, we crawled profiles to collect public information that are available to everyone. However, Facebook, as most OSNs operators do, protects this data from exhaustive crawling by implementing a plethora of anti-crawler techniques. For instance, it implements a request rate limit that, if exceeded, generates a CAPTCHA to be solved. To bypass this restriction and to be cautious not to DoS the system, we set a very slow request frequency (1 per minute). In addition, we distributed our crawler on 6 different machines that were geographically spread. In addition, it is worth noting that one of the trickiest countermeasures that Facebook implements to prevent easy crawling is the rendering of the web page. In particular, rather than sending a simple HTML page to the client browser, Facebook embeds HTML inside JavaScript, thus, the received page is not a valid HTML page but a JavaScript code that has to be interpreted. Unfortunately, most publicly available crawling libraries do not interpret JavaScript. Thus, we developed our own lightweight web browser, based on the Qt Port of WebKit [1], which is capable of interpreting JavaScript. This allows our crawler to be served with easy-to-parse HTML page.

[3]available at:`http://www.facebook.com/directory/`

### 5.2. A Facebook Application to Collect Private Attributes

We developed a Facebook application to gather private attributes from users. The application was distributed to many of our colleagues and friends on Facebook, and was surprisingly used by more users than expected. Users volunteered to install the application, and hence, their private information was collected by our tool. We collected private attributes from 4012 profiles out of which 2458 profiles have at least one music interest. These anonymized private profiles, collected from April 6 to April 20 in 2011, represent our private dataset (called VolunteerProfiles).

The usage of this dataset is motivated by our need to understand how data availability varies between public and private datasets, and to verify whether it impacts the results of our algorithm.

### 5.3. Ethical and Legal Considerations

In order to comply with legal and ethical aspects in crawling online social networks data, we were cautious not to inadvertently DoS the Facebook infrastructure (as mentioned in Section 5.1). Also cautionary measures were taken to prevent our crawler from requesting off-limit information. In other words, our crawler is compliant with the Robots Exclusion Protocol [2]. Even though we accessed publicly available information, we anonymized the collected data by removing user names and all information which were irrelevant to our study.

The Facebook application needed more sanitization to ensure users' anonymity. The reader might refer to the 'Disclosure and Privacy Policy' of the application[4] for more information.

[4]available at `http://apps.facebook.com/social_privacy/`

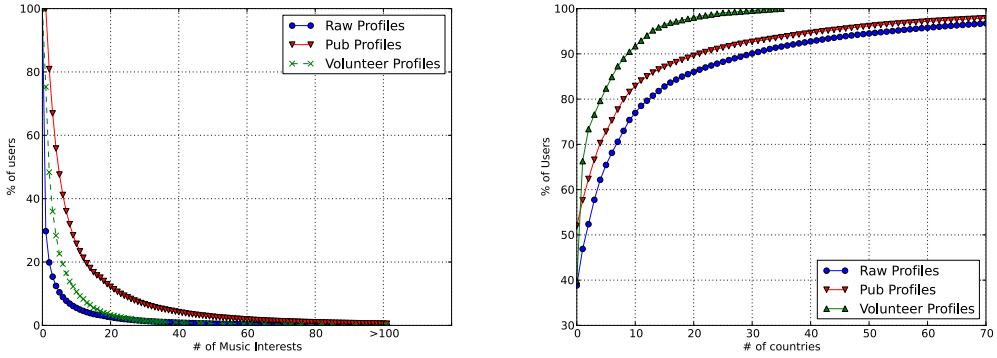


Figure 2: Left: Complementary Cumulative Distribution Function of Music Interests. Right: Cumulative Distribution Function (CDF) of Country-level Locations (retrieved from the $CurrentCity$ attribute)

### 5.4. Dataset Description

In the following, we provide statistics that describe the datasets used in this study. First, Table 1 summarizes the statistics about the availability of attributes in the three datasets (i.e., in RawProfiles, PubProfiles and VolunteerProfiles).

| Attributes | $Raw(\%)$ | $Pub(\%)$ | $Volunteer(\%)$ |
|---|---|---|---|
| Gender | 79 | 84 | 96 |
| Interests | 57 | 100 | 62 |
| Current City | 23 | 29 | 48 |
| Looking For | 22 | 34 | - |
| Home Town | 22 | 31 | 48 |
| Relationship | 17 | 24 | 43 |
| Interested In | 16 | 26 | - |
| Birth date | 6 | 11 | 72 |
| Religion | 1 | 2 | 0 |

Table 1: The availability of attributes in our datasets.

We observe that Gender is the most common attribute that users publicly reveal. However, three attributes that we want to infer are largely kept private. The age is the information that users conceal the most (89% are undisclosed in PubProfiles). Comparing the availability of the attributes in PubProfiles and VolunteerProfiles is enlightening. We can clearly note that users tend to hide their attribute values from public access even though these attributes are frequently provided (in their private profiles). For instance, the birth date is provided in more than 72% in VolunteerProfiles, whereas it is rarely available in PubProfiles (only 1.62% of users provide their full birth date). The current city is publicly revealed in almost 30% of the cases, whereas half of all volunteers provided this data in their private profile. Recall that the attributes we are interested in are either binary (Gender, Relationship) or multi-valued (Age, Country-level location). Finally, note that, as it is shown in Table 1, the public availability of attributes in PubProfiles and in RawProfiles are roughly similar.

Also note that the availability of interests slightly changes from RawProfiles (57%) to VolunteerProfiles (62%), yet still relatively abundant. This behavior might have at least two explanations: (1) by default, Facebook sets Interest to be a public attribute, (2) users are more willing to reveal their interests compared to other attributes. Figure 2 (left) depicts the complementary CDF of music interests publicly revealed by users in the three datasets. Note that more than 30% of RawProfiles profiles reveal at least one music interest. Private profiles show a higher ratio which is more than 75%.

Figure 2 (right) plots the cumulative distribution of the country-level locations of users in our datasets. The three curves show that a single country is over-represented, and that a large fraction of users' locations is represented only by a few countries. Independently from the dataset, 40% of users come from a single country, and the top 10 countries represent more than 78% of users. The gentler slope of the curves above 10 countries indicates that other countries are more widely spread across the remaining profiles. Notably, the number of countries appearing in VolunteerProfiles shows that the distribution does not cover all countries in the world. In particular, our volunteers only come from less than 35 different countries. Nevertheless, we believe that VolunteerProfiles still fits for purpose because the over-representation shape of location distributions is kept, and illustrated by the Facebook statistics [4] in general (more than 50% of users come from only 9 countries). Motivated by this over-representation in our datasets, we validate our inference technique in Section 6.2 on users that come from the top 10 countries (following the Facebook statistics).

| *Attribute* | *Overall marginal distribution (OMD)* | | *Inference accuracy on* VolunteerProfiles | |
|---|---|---|---|---|
| | PubProfiles | *Facebook statistics* | PubProfiles OMD | *Facebook statistics OMD* |
| Gender | 62% (Female) | 51% (Male) | 39.3% | 60.7% |
| Relationship | 55% (Single) | Unknown | 36.7% | 50%[4] |
| Age | 50% (18-25) | 26.1% (26-34) | 33.9% | 57.9% |
| Country | 52% (U.S) | 23% (U.S) | 2.3% | 2.3% |

Table 2: Baseline inference using different marginal distributions. Inference of VolunteerProfiles based on Facebook OMD is better than PubProfiles OMD.

## 6. Experimentation Results and Validation

In the following, we validate our interest-based inference technique using both VolunteerProfiles and PubProfiles. We evaluated the correctness of our algorithm in terms of inference accuracy, i.e. the fraction of successful inferences and the total number of trials. An inference is successful if the inferred attribute equals to the real value. In particular, for both PubProfiles and VolunteerProfiles datasets and for each attribute to be inferred, we select users that provide the attribute and then we compute the inference accuracy: we hide each user's attribute, compute the nearest neighbors of the user, do a majority voting as described in Section 4.5.1, and then verify whether the inference yields the real attribute.

Before discussing our validation results, in the following, we introduce a maximum likelihood-based inference technique that we consider as a baseline technique with which we compare our method.

### 6.1. Baseline Inference Technique

Without having access to any friendship and/or community graph, an adversary can rely on the marginal distributions of the attribute values. In particular, the probability of value val of a hidden attribute $x$ in any user's profile $u$ can be estimated as the fraction of users who have this attribute value in dataset $U$:

$$P(u.x = \mathsf{val}|U) = \frac{|\{v \,|\, v.x = \mathsf{val} \wedge v \in U\}|}{|U|}$$

Then, a simple approach to infer an attribute is to guess its most likely value for all users (i.e., the value $x$ for which $P(u.x = \mathsf{val}|U)$ is maximal).

To compute $P(u.x = \mathsf{val}|U)$, an adversary can crawl a set of users and then derive the Overall Marginal Distribution (OMD) of an attribute $x$ from the crawled dataset (more precisely, $U$ is the subset of all crawled users who published that attribute). However, this OMD is derived from public attributes (i.e., $U$ contains only publicly revealed attributes), and hence, may deviate from the real OMD which includes both publicly revealed and undisclosed attributes.

[4]Using random guessing instead of maximum likelihood decision

To illustrate the difference, consider Table 2 that compares the real OMD of the four attributes to be inferred, as provided by Facebook statistics (composed of both private and public attributes [4]), with the OMD derived from our public dataset PubProfiles. The two distributions suggest different predominant values which highly impacts the inference accuracy when the guess is based on the most likely attribute value. For instance, PubProfiles conveys that the majority of Facebook users are female which contradicts Facebook statistics (with a significant difference of 11%). Similarly, the age of most users according to PubProfiles is between 18 and 25-years old, while the predominant category of ages, according to Facebook, is 26-34.

In fact, all public datasets (e.g., PubProfiles) are biased towards the availability of attributes (not to be confused with the bias in sampling discussed in Section 5.1). Recall that, as shown in Table 1, some attributes (in particular Age, Relationship status and Country) are publicly available for only a small fraction of users (see the PubProfiles column). Put simply, the difference between the two OMDs is mainly due to the mixture of private and public attributes in Facebook statistics and the absence of private attributes in PubProfiles. Whether revealing attributes is driven by some sociological reasons or others is beyond the scope of this paper.

To illustrate how the bias towards attribute availability impacts inference accuracy, we conduct two experiments. First, we infer the attributes in VolunteerProfiles using the OMD derived from PubProfiles. In the second experiment, we infer the same attributes using the OMD computed from Facebook statistics. As shown in Table 2, the second approach always performs better. The results show that using the Facebook statistics we obtain an inference accuracy gain of 21% for the gender and 25% for the age. Since Facebook does not provide statistics about the relationship status of their users, we used random guessing instead (i.e., we randomly chose between single and married for each user). Surprisingly, even random guessing outperforms the maximum likelihood-based approach using PubProfiles OMD. Therefore, we conclude that the maximum likelihood-based inference performs better when we use the OMD derived from Facebook statistics. Accordingly, in our performance evaluation, we also used this in our baseline inference tech-

nique.

Finally, note that previous works [27, 18] computed the inference accuracy using private data (i.e., their dataset is a crawl of a community, and thus, they could access all attributes that can only be seen by community members). Hence, these results are obtained with different attacker model, and the assumption that 50% of all attributes are accessible, as suggested in [27], is unrealistic in our model.

### 6.2. Experiments

In order to validate our interest-based inference technique, we follow two approaches. First, for each attribute, we randomly sample users from PubProfiles such that the sampled dataset has the same OMD as the real Facebook dataset [4]. Then, we measure the inference accuracy on this sampled dataset. Second, we test our technique on the VolunteerProfiles dataset where both private and public attributes are known. Since we know the attribute values in the collected profiles, we can check if the inference is successful or not. In particular, we infer four attributes in both approaches: Gender, Relationship status, Age, and the Country of current location. We run experiments to infer an attribute $a$ in PubProfiles as follows:

1. From all users that provide $a$ in PubProfiles, we randomly sample a set of users (denoted by $S$) following the OMD of Facebook. The size of $S$ for each attribute is tailored by (i) Facebook OMD and (ii) the number of available samples in PubProfiles. Table 3 shows the size of $S$.
2. For this sampled set, we compute the inference accuracy as it has been described in Section 6.2.
3. We repeat Steps 2 and 3 fifteen times and compute the average of all inference accuracy values (Monte Carlo experiment).

For VolunteerProfiles we proceed as for PubProfiles, but since the attributes are a mix of public and private attributes, there is no need to do sampling, and we skip Step 1.

| Attribute | Size of $S$ |
|---|---|
| Gender | 1000 |
| Relationship | 400 |
| Country | 1000 |
| Age | 105 |

Table 3: Size of $S$

**Parameter Estimation** Recall from Section 4.5.1 that our algorithm is based on majority voting. Hence, estimating the number of neighbors that provides the best inference accuracy for each attribute is essential. Figure 3 depicts the inference accuracy in function of the number of neighbors. This figure clearly shows that each attribute has a specific number of neighbors that results in the best inference accuracy. Note that, as discussed at the beginning of this section, we rely on repeated random sampling to compute the results, and hence, the computed parameters are independent from the input data. Age inference requires two neighbors. This can be explained by the limited number of users that disclose their age which causes the IFV space to be very sparse: the more neighbors we consider the more likely it is that these neighbors are far and have different attribute values. For other attributes, the optimal number of neighbors is between 3 and 5. We tested different IFV sizes (i.e., $k$ the number of topics). Notably, best results were achieved with $k = 100$. In the sequel, we will use these estimated numbers of neighbors as well as $k = 100$ which yield the best inference accuracy.

| Attribute | Baseline | Random guess | IFV Inference |
|---|---|---|---|
| Gender | 51% | 50% | 69% |
| Relationship | 50% | 50% | 71% |
| Country | 41% | 10% | 60% |
| Age | 26% | 16.6% | 49% |

Table 4: Inference Accuracy of PubProfiles

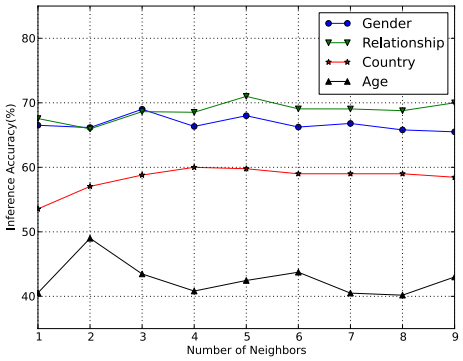


Figure 3: Correlation between Number of Neighbors and Inference accuracy

Table 4 provides a summary of the results for PubProfiles. The information leakage can be estimated to 20% in comparison with the baseline inference. Surprisingly, the amount of the information is independent from the inferred attribute since the gain is about 20% for all of them. These results show that music interest is a good predictor of all attributes.

| Attribute \ Inferred | Male | Female |
|---|---|---|
| Male | 53% | 47% |
| Female | 14% | 86% |

Table 5: Confusion Matrix of Gender

**Gender Inference** Table 4 shows that the gender can be inferred with a high accuracy even if only one music interest is known in the PubProfiles. Our algorithm performs 18% better than the baseline. Recall that the baseline guesses male for all users (Table 2). To compare the inference accuracy for both males and females, we computed the confusion matrix in Table 5. Surprisingly, memale inference is highly accurate (86%) with a low false negative rate (14%). However, it is not the case for male inference. This behavior can be explained by the number of female profiles in our dataset. In fact, females represent 61.41% of all collected profiles (with publicly revealed gender attribute) and they were subscribed to 421685 music interests. However, males share only 273714 music interests which represents 35% less than woman. Hence, our technique is more capable of predicting females since the amount of their disclosed (music) interest information is larger compared to males. This also confirms that the amount of disclosed interest information is correlated with inference accuracy.

| Attribute \ Inferred | Single | Married |
|---|---|---|
| Single | 78% | 22% |
| Married | 36% | 64% |

Table 6: Confusion Matrix of Relationship

**Relationship Inference** Inferring the relationship status (married/single) is challenging since less than 17% of crawled users disclose this attribute showing that it is highly sensitive. Recall that, as there is no publicly available statistics about the distribution of this attribute, we do random guessing as the baseline (having an accuracy of 50%). Our algorithm performs well with 71% of good inference for all users in PubProfiles. As previously, we investigate how music interests are a good predictor for both single and married users by computing the confusion matrix (Table 6). We notice that single users are more distinguishable, based on their IFV, than married ones. The explanation is that single users share more interests than married ones. In particular, a single user has an average of 9 music interests whereas a married user has only 5.79. Likewise in case of gender, this confirms that the amount of disclosed interest information is correlated with inference accuracy.

**Country of Location Inference** As described in Section 5, we are interested in inferring the users' location in the top 10 countries in Facebook. Our approach can easily be extended to all countries, however, as shown by Figure 2, more than 80% of users in PubProfiles belong to 10 countries and these countries represent more than 55% of all Facebook users according to [4]. As the number of users belonging to the top 10 countries is very limited in VolunteerProfiles, we do not evaluate our scheme on that dataset. Table 4 shows that our algorithm has an accuracy of 60% on PubProfiles with 19% increase compared to the baseline (recall that, following Table 2, the baseline gives U.S. as a guess for all users). Figure 4 draws the confusion matrix[6] and gives more insight about the inference accuracy. In fact, countries with a specific (regional) music have better accuracy than others. Particularly, U.S. has more than 94% of correct inference, Philippine 80%, India 62%, Indonesia 58% and Greece 42%. This highlights the essence of our algorithm where semantically correlated music interests are grouped together and hence allow us to extract users interested in the same topics (e.g., Philippine music). Without a semantic knowledge that specifies the origin of a singer or band this is not possible. As for Gender and relationship, the number of collected profiles can also explain the incapacity of the system to correctly infer certain countries such as Italy, Mexico or France. In particular, as shown in Table 7, the number of users belonging to these countries is very small. Hence, their interests may be insufficient to compute a representative IFV which yields poor accuracy.

| Att \ Inferred | 13-17 | 18-24 | 25-34 | 35+ |
|---|---|---|---|---|
| 13-17 | 58.33% | 30% | 11.6% | 0% |
| 18-24 | 17% | 67% | 3.4% | 1.3% |
| 25-34 | 15.38% | 46.15% | 38.4% | 0% |
| 35+ | 0% | 100% | 0% | 0% |

Table 8: Confusion Matrix of Age Inference

**Age Inference** Finally, we are interested in inferring the age of users. To do that, we created five age categories[7] that are depicted in Table 8. Recall that the baseline technique always predicts the category of 26 and 34 years for all users. Table 4 shows that our algorithm performs 23% better than the baseline attack. Note that our technique gives good results despite that only 3% of all users provide their age (3133 users in total) in PubProfiles. We investigate how music interests are correlated with the age bin by computing

[6] We removed Brazil since all its entries (2) were wrongly inferred. This is caused by the small number of Brazilians in our dataset.

[7] We created six categories but since in PubProfiles we have only few users in the last 3 bins, we merge them together. For VolunteerProfiles we have six bins.

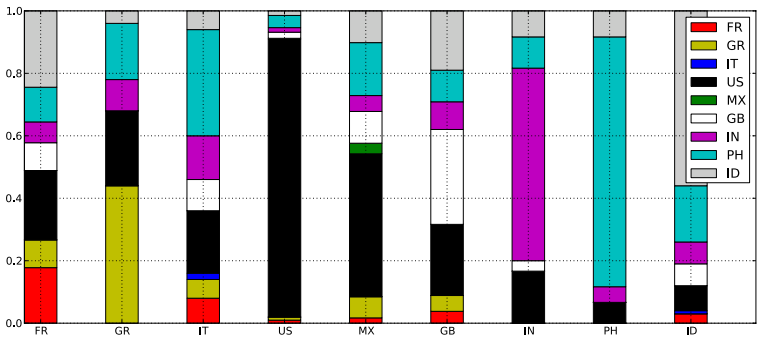


Figure 4: Confusion Matrix of Country Inference

| Country | % of users |
|---|---|
| US | 71.9% |
| PH | 7.80% |
| IN | 6.21% |
| ID | 5.08% |
| GB | 3.62% |
| GR | 2.32% |
| FR | 2.12% |
| MX | 0.41% |
| IT | 0.40% |
| BR | 0.01% |

Table 7: Top 10 countries distribution in PubProfiles

the confusion matrix in Table 8. We find that, as expected, most errors come from falsely putting users into their neighboring bins. For instance, our method puts 30% of 13-18 years old users into the bin of 18-24 years. However, note that fewer bins (such as teenager, adult and senior) would yield better accuracy, and it should be sufficient to many applications (e.g., for targeted ads). Observe that we have an error of 100% for the last bin. This is due to the small number of users (3 in PubProfiles) who belong to this bin (we cannot extract useful information and build a representative IFV for such a small number of users).

#### 6.2.1 VolunteerProfiles Inference

| Attribute | Baseline | Random guess | IFV Inference |
|---|---|---|---|
| Gender | 51% | 50% | 72.5% |
| Relationship | 50% | 50% | 70.5% |
| Age | 26% | 16.6% | 42% |

Table 9: Inference Accuracy for VolunteerProfiles

As a second step to validate our IFV technique, we perform inference on VolunteerProfiles. Table 9 shows that our algorithm also performs well on this dataset. Notice that Age inference is slightly worse than in PubProfiles. Recall from Section 6.2 that we had only a few users in the last three bins in PubProfiles, and hence, we merged these bins. However, in VolunteerProfiles, we have enough users and we can have 6 different age categories. This explains the small difference in inference accuracy between VolunteerProfiles and PubProfiles. Regarding other attributes, the accuracy is slightly worse for Relationship (-0.5%) and better for Gender (+3.5%). This small variation in inference accuracy between PubProfiles and VolunteerProfiles demonstrates that our technique has also good results with users having *private* attributes: in PubProfiles, we could compute the inference accuracy only on users who published their attribute values, while in VolunteerProfiles, we could also test our method on users hiding their attributes.

## 7. Discussion

**Topic modeling** We used LDA for semantic extraction. Another alternative is to use Latent Semantic Analysis (LSA) [17]. As opposed to LDA, LSA is not a generative model. It consists in extracting a spatial representation for words from a multi-document corpus by applying singular value decomposition. However, as pointed out in [12], spatial representations are inadequate for capturing the structure of semantic association; LSA assumes symmetric similarity between words which is not the case for a vast majority of associations. One classical example given in [12] involves China and North Korea: Griffiths et al. noticed that, generally speaking, people have always the intuition that North Korea is more similar to China than China to North Korea. This problem is resolved in LDA where $P(\text{occurence of } word1 | \text{occurence of } word2) \neq P(\text{occurence of } word2 | \text{occurence of } word1)$. In addition, [12] showed that LDA outperforms LSA in terms of drawing semantic correlations between words.

**Collaborative Filtering** Our algorithm is based on discovering latent correlations between user interests in order to cluster users. An alternative approach could be to employ model-based collaborative filtering (MBCF) that avoids using semantic-knowledge. In MBCF, each user is represented by his interest vector. The size of this vector equals the number of all defined interest names, and its coordinates are defined as follows: a coordinate is 1 if the user has the corresponding interest name, otherwise it is 0. Since interest names are user-generated, the universe of all such names, and hence the vector size can be huge. This negatively impacts the performance.

In particular, collaborative filtering suffers from a "cold-start" effect [24], which means that the system cannot draw correct inferences for users who have insufficient information (i.e., small number of interests). Recall from Section 5 that 70% of users in PubProfiles have less than 5 interests and it is more than 85% in RawProfiles. Hence, the sparseness of users' interest vectors is very high (the average density[8] is 0.000025). Moreover, [8] has studied the effect of cold-start in recommendation systems (for both item-based and collaborative-based) on real datasets gathered from two IP-TV providers. Their results show that a well-known CF algorithms, called SVD [20], performs poorly when the density is low (about 0.0005) with a recall between 5% and 10%. Additionally, the number of new users and interests is ever growing (on average, 20 millions new users joined Facebook each month in the first half of 2011 [4]). This tremendous number of new users and interests keeps the system in a constant cold-start state. In addition, users, in MBCF typically evaluate items using a multi-valued metric (e.g., an item is ranked between 1 and 5) but it must be at least binary (e.g. like/dislike), whereas in our case, only "likes" (interests) are provided. In fact, the lack of an interest $I$ in a user profile does not mean that the user is not interested in $I$, but he may simply not have discovered $I$ yet. In these scenarios, when users only declare their interests but not their disinterest, MBCF techniques (e.g. SVD [20]) are less accurate than nearest neighbor-like approaches that we employed [16].

**OSN independence** One interesting feature of our technique is that it is OSN independent. In particular, it does not rely on any social graph, and the input data (i.e. interest names) can be collected from any other source of information (e.g., deezer, lastfm, or any other potential sources).

**No need for frequent model updates (stability)** One may argue that our LDA model needs frequent updates since user interests are ever-growing. Nevertheless, recall from Section 4.2 that our technique uses the parent topics of the user interests (according to Wikipedia) to augment the semantics knowledge of each interest. There are substantially fewer higher-level parent categories than leaf categories in the Wikipedia hierarchy, and they change less frequently. Thus, there is no need to update the LDA model, unless the considered interest introduces a new parent category in the running model. Hence, our approach is more stable than MBCF; once the IFV vector is extracted and similarity is computed, we can readily make inference *without* having to retrain the system.

[8]The density of this vector is the number of coordinates equal one divided by the vector size.

**Targeted advertising and spam** Using our technique, advertisers could automatically build online profiles with high accuracy and minimum effort *without* the consent of users. Spammers could gather information across the web to send targeted spam. For example, by matching a user's Facebook profile and his email address, the spammer could send him a message containing ads that are tailored to his inferred geo-localization, age, or marital status.

**Addressing possible limitations** First, we only tested our approach on profiles that provide music interests. Even with this limitation, our results show the effectiveness of our technique in inferring undisclosed attributes. In addition, we only based our method on user interests and did not combine it with any other available attributes (e.g. gender or relationship) to improve inference accuracy. We must emphasize that our main goal was to show information leakage through user interests rather than developing a highly accurate inference algorithm. Considering other interests (e.g. movies, books, etc.) and/or combining with different available attributes can be a potential extension of our scheme which is left for future work. Second, we demonstrated our approach using an English-version of Wikipedia. However, our approach is not restricted to English, since Wikipedia is also available in other languages. Finally, we encountered few examples that denotes a non-interest (or "dislike"). In particular, we observed interests that semantically express a dislike for a group, or an ideology. For instance, an interest can be created with the title "I hate Michael Jackson". Our semantics-driven classification will falsely identify the users having this interest as "Michael Jackson" fans. However, as a minority of users are expected to use such a strategy to show their non-interest, this has a small impact on our approach. Additionally, one might integrate Applied Linguistics Techniques to handle such peculiar cases and filter out dislikes.

## 8 Conclusion

This paper presents a semantics-driven inference technique to predict private user attributes. Using only Music Interests that are often disclosed by users, we extracted unobservable Interest topics by analyzing the corpus of Interests, which are semantically augmented using Wikipedia, and derived a probabilistic model to compute the belonging of users to each of these topics. We estimated similarities between users, and showed how our model can be used to predict hidden information. Therefore, on-line services and in particular OSNs should raise the bar of privacy protections by setting a restrictive by-default behavior, and explicitly hide most user information.

## References


[1] Qt port of webkit: an open source web browser engine. http://trac.webkit.org/wiki/QtWebKit.

[2] Robots Exclusion Protocol. RobotsExclusionProtocol, 1996.

[3] OpenCyc. http://www.opencyc.org/, 2006.

[4] Facebook Statistics. http://gold.insidenetwork.com/facebook/facebook-stats/, 2011.

[5] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 181–190, New York, NY, USA, 2007. ACM.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[7] W.-Y. Chen, J.-C. Chu, J. Luan, H. Bai, Y. Wang, and E. Y. Chang. Collaborative filtering for orkut communities: discovery of user latent behavior. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 681–690, New York, NY, USA, 2009. ACM.

[8] P. Cremonesi and R. Turrin. Analysis of cold-start recommendations in IPTV systems. In *RecSys '09: Proceedings of the third ACM conference on Recommender systems*, pages 233–236, New York, NY, USA, 2009. ACM.

[9] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407, 1990.

[10] C. Fellbaum, editor. *WordNet An Electronic Lexical Database*. The MIT Press, Cambridge, MA ; London, May 1998.

[11] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7, December 2005.

[12] T. L. Griffiths, J. B. Tenenbaum, and M. Steyvers. Topics in semantic representation. *Psychological Review*, 114:2007, 2007.

[13] J. He, W. W. Chu, and Z. (victor Liu. Inferring privacy information from social networks. In *IEEE International Conference on Intelligence and Security Informatics*, 2006.

[14] T. Hofmann. Probabilistic Latent Semantic Analysis. In *Proceedings of Uncertainty in Artificial Intelligence, UAI*, 1999.

[15] M. Kurant, M. Gjoka, C. T. Butts, and A. Markopoulou. Walking on a Graph with a Magnifying Glass. In *Proceedings of ACM SIGMETRICS '11*, San Jose, CA, June 2011.

[16] S. Lai, L. Xiang, R. Diao, Y. Liu, H. Gu, L. Xu, H. Li, D. Wang, K. Liu, J. Zhao, and C. Pan. Hybrid recommendation models for binary user preference prediction problem. In *KDD Cup*, 2011.

[17] T. K. Landauer and S. T. Dumais. Solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 1997.

[18] J. Lindamood and M. Kantarcioglu. Inferring Private Information Using Social Network Data. Technical report, University of Texas at Dallas, 2008.

[19] Z. Liu, Y. Zhang, E. Y. Chang, and M. Sun. Plda+: Parallel latent dirichlet allocation with data placement and pipeline processing. *ACM Transactions on Intelligent Systems and Technology, special issue on Large Scale Machine Learning*, 2011. Software available at http://code.google.com/p/plda.

[20] D. B. Michael. Learning collaborative information filters, 1998.

[21] D. Milne. An open-source toolkit for mining wikipedia. In *Proc. New Zealand Computer Science Research Student Conf*, 2009.

[22] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel. You are who you know: Inferring user profiles in online social networks.

[23] J. Puzicha, T. Hofmann, and J. Buhmann. Non-parametric similarity measures for unsupervised texture segmentation and image retrieval. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 267 –272, jun 1997.

[24] A. I. Schein, A. Popescul, L. H., R. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260. ACM Press, 2002.

[25] A. L. Traud, P. J. Mucha, and M. A. Porter. Social Structure of Facebook Networks. 2011.

[26] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A practical attack to de-anonymize social network users. In *31st IEEE Symposium on Security and Privacy, Oakland, California, USA*, 2010.

[27] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *In WWW*, 2009.

[28] E. Zheleva, J. Guiver, E. M. Rodrigues, and N. Milic-Frayling. Statistical models of music-listening sessions in social media. In *In WWW*, 2010.